4197 Combination Lock

A combination lock consists of a circular dial, which can be turned (clockwise or counterclockwise) and is embedded into the ”fixed” part of the lock. The dial has N evenly spaced ”ticks”. The ticks are numbered from 0 to $N-1$, increasing in the clockwise direction. The fixed part of the lock has a ”mark” which always ”points to” a particular tick on the dial. Of course, the mark points to different ticks as the dial is turned.

The lock comes with three code numbers $T1$, $T2$, $T3$. These are non-negative integers and each of them is less than N. No two of the three are the same.

The lock is opened in three stages of operations:

1. Turn the dial clockwise exactly two full revolutions, and continue to turn it clockwise until the mark points to tick $T1$.

2. Turn the dial one full revolution counterclockwise and continue to turn it counterclockwise until the mark points to tick $T2$.

3. Turn the dial clockwise until the mark points to tick $T3$. The lock should now open.

You must find the maximum possible number of ticks the dial must be turned in order to open the lock. The number of ticks turned is defined to be the sum of the ticks turned in the three stages outlined above, and is always positive regardless of direction.

Input

The input file consists of a number of test cases, one test case per line. Each line of the input file contains four integers: N, $T1$, $T2$, $T3$, in this order, separated by blank spaces. The integer N is a multiple of 5, $25 \leq N \leq 100$. The numbers $T1$, $T2$ and $T3$ satisfy the constraints stated under the description above. The input will be terminated by a line with $N = T1 = T2 = T3 = 0$.

Output

For each test case, print the maximum possible number of ticks the dial must be turned in order to open the lock. Print each on its own line.

Sample Input

```
80 20 40 50
80 10 79 12
0 0 0 0
```

Sample Output

```
409
455
```